

# **SDRAM (synchronní DRAM)**

## **Cíl přednášky:**

- Shrnout předcházející techniky řízení paměti.
- Prezentovat techniku SDRAM, postihnout její výrazné rysy a odlišnosti od předcházejících typů.
- Shrnout získané informace.

## **Shrnutí technik FPM, EDO, BEDO**

- Technika FPM DRAM umožňuje realizovat čtení tak, že je nastaven ROW na určitou hodnotu a přičítá se hodnota COLUMN.
- Technika EDO – prodlužuje se doba, po níž jsou přečtená data k dispozici na datové sběrnici => řadič paměti má více času na to, aby předal data přes sběrnici do procesoru.
- FPM i EDO – komunikace je asynchronní a odvíjí se od signálů CAS, RAS.
- Synchronní DRAM – operace jsou synchronizovány (synchronizace komunikace mezi řadičem paměti a pamětí – synchronizační pulsy – příkazy, data a adresa jsou synchronizovány nástupnou hranou synchronizačního signálu => zesynchronizování činnosti paměti s ostatními komponentami).
- Další výrazný rys – paměť je řízena příkazy, nikoliv signály jako je tomu u FPM/EDO/BEDO.
- Synchronizační pulsy jsou generovány řadičem paměti (systémové hodiny jsou přes řadič paměti přenášeny do SDRAM).
- FPM/EDO/BEDO dovolují napájení napětím 5V/3,3V, synchronní DRAM jsou napájeny

napětím 3,3V nedovolují řízení signály na úrovni 5V.

- Řadiče paměti – jsou většinou schopny řídit paměti FPM/EDO/BEDO i SDRAM.

## **Signály řadiče SDRAM**

- Při návrhu řadiče SDRAM se vycházelo z faktu, že existuje řadič paměti FPM, EDO, BEDO a má definované rozhraní.
- Paměť SDRAM je řízena kombinací signálů RAS, CAS a WE, které představují kód příkazu.
- Každou nástupnou hranou signálu CLK se tyto signály nahrávají do vnitřní logiky (záchytné klopné obvody – latch) a pak jsou dekódovány, SDRAM pak provede požadovanou funkci.
- Časové relace mezi signály CS, RAS, CAS a WE nehrají roli, důležitý je stav těchto signálů v okamžiku nástupné hrany, to platí i o signálech na datové a adresové sběrnici.
- Řadič paměti vkládá do rozhraní s pamětí signál CLK, jeho nástupnou hranou se vkládají údaje do paměti – řízení i data.
- Signál CKE (Clock Enable) – možnost zablokování CLK. Využívá se při POWER-

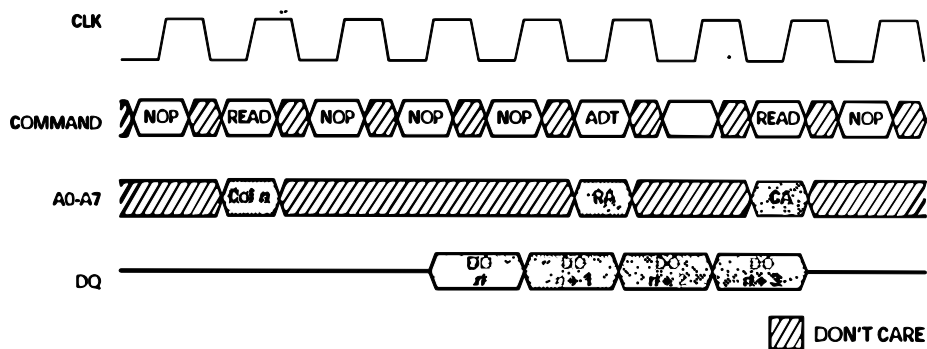
DOWN (vypnutí), SELF-REFRESH (autorefreši).

- Signál CS (Chip Select) – uvolňuje činnost čipu, tzn. např. vložení dat na datovou sběrnici. Je proto využíván také klasicky na výběr paměťového čipu na základě adresy.
- Signály DQ – data.
- Signál DQM – zablokování výstupů.
- Registr MODE (režim) – řízení funkcí SDRAM, jako je:

Objem dat přenášený v nárazovém režimu (burst length) při čtení nebo zápisu dat – počet sloupců (columns).

CAS latency – určuje, za jak dlouho po vygenerování příkazu READ nebo WRITE budou na výstupech DQ adresovaná data k dispozici (typické hodnoty jsou 2 – 3).

# Časový diagram komunikace s pamětí SDRAM

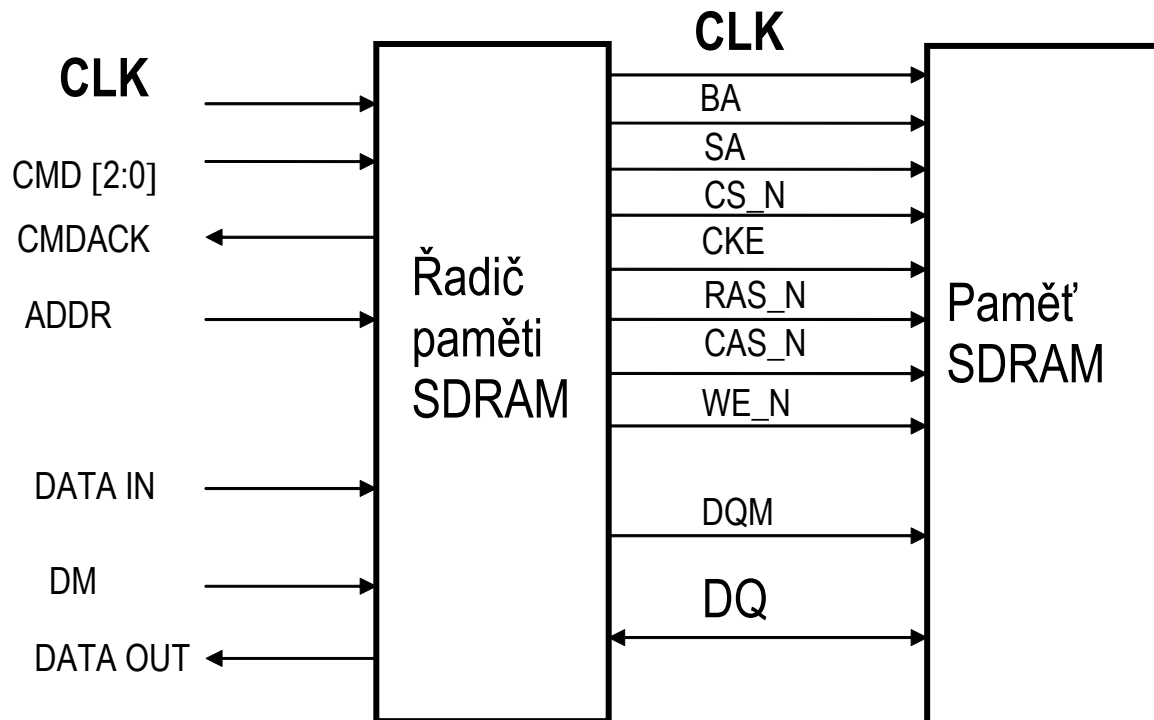


## Čtení – paměť typu SDRAM

**Důležité:** Zatímco u pamětí DRAM (FPM, EDO, BEDO) je činnost asynchronní a je řízena signálem CAS, u paměti SDRAM je tomu zcela jinak – paměť je řízena příkazy (3 bity), jedním z bitů je také signál CAS.

V komunikaci mezi řadičem a pamětí hraje důležitou roli synchronizační signál – v okamžiku nástupné hrany synchronizačního signálu jsou vyhodnocovány signály rozhraní (především kód příkazu na vodičích CAS, RAS, WE).

Mezi příkazy, na základě nichž se má provést konkrétní činnost, se vkládají příkazy NOP.



- **RAS\_N/CAS\_N/WE\_N/CS\_N:** signály jsou aktivní na úrovni L.

### Popis signálů:

- **CLK**  
Systémové hodiny.
- **RESET**  
Systémové nulování.
- **ADDR**  
Adresa paměti pro operace čtení/zápis, příp.pro přenos jiných údajů.

- **CMD** (command)  
Kód příkazu a zároveň požadavek o provedení příkazu (změnou z kódu NOP na jinou kombinaci).
- **CMDACK** (command acknowledge)  
Potvrzení přijatého příkazu.
- **DATA IN**  
Vstupní data.
- **Výstupní data**  
Výstupní data.
- **DM**  
Maska uplatněná na jednotlivé byty datové sběrnice.
- **SA**  
Adresová sběrnice.
- **BA [1 : 0]**

Adresa banku (určení, pro který z banků mají být příkazy ACT, RD, WR, PCH použity).

- **CS\_N [1 : 0]**  
Signály Chip select pro jednotlivé banky.
- **CKE**  
Clock enable.
- **RAS\_N (Row Address Strobe)**  
Dvě funkce:  
Jeden ze 3 bitů kódu příkazu.  
Tímto signálem je také do SDRAM vkládána adresa řádku.
- **CAS\_N (Column Address Strobe)**  
Dvě funkce:  
Jeden ze 3 bitů kódu příkazu.  
Tímto signálem je také do SDRAM vkládána adresa sloupce.
- **WE\_N (Write Enable)**  
Jeden ze 3 bitů kódu příkazu.



- **DQ**

Datová sběrnice SDRAM.

- **DQM**

Maska dat přenášených do SDRAM při zápisu dat do SDRAM.

**Tab. 1: Příkazy přijímané řadičem paměti a jejich kódy**

<b>Příkaz CMD[0:2]</b>	<b>Kód</b>	<b>Popis činnosti</b>
NOP	000	No operation.
READA	001	Čti SDRAM + auto precharge
WRITEA	010	Piš do SDRAM + auto precharge
REFRESH	011	refreš
PRECHARGE	100	Uzavři všechny banky
LOAD_MODE	101	Zapiš do registru mode
LOAD_REG1	110	Zapiš do konfiguračního registru R1
LOAD_REG2	111	Zapiš do konfiguračního registru R2

**Tab. 2: Příkazy generované řadičem pro SDRAM**

Příkaz	Zkratka	RASN	CASN	WEN
No operation	NOP	H	H	H
Activate	ACT	L	H	H
Read	RD	H	L	H
Write	WR	H	L	L
Burst terminate	BT	H	H	L
Precharge	PCH	L	H	L
Autorefresh	ARF	L	L	H
Load mode register	LMR	L	L	L

- **Činnost řadiče SDRAM při provádění příkazu:**
  1. Na vstup řadiče SDRAM je vložen kód příkazu (např. čtení) – signály CMD[2:0].
  2. Řadič SDRAM příkaz přijme a potvrdí přijetí příkazu generováním signálu CMDACK (Command Acknowledge).
  3. Současně s potvrzením příkazu zahájí provedení příkazu v SDRAM (generováním příslušných signálů).
  4. Na vstup řadiče SDRAM je vložen kód příkazu NOP – signály CMD[2:0] – s hranou CLK po přijetí CMDACK.

- Pokud se od řadiče SDRAM a paměti nevyžaduje žádná činnost, pak je na CMD[2:0] příkaz NOP (tzn. na vstupu řadiče SDRAM). Každá změna v kódu příkazu je rozpoznána a příkaz se provede.
- Totéž platí o rozhraní mezi řadičem SDRAM a samotnou SDRAM.
- Pokud se má z paměti číst nebo do ní zapisovat, tak řadič paměti rozdělí příkaz na dvě části: ACT a pak RD nebo WR.
- Funkce vodičů SA[11:0] mezi řadičem a pamětí:

Příkaz ACT: informace na SA[11:0] – v tomto případě řádek (row) - se vloží do registru paměti.

Příkaz RD/WR: informace na SA[11:0] – v tomto případě sloupec (column) - se vloží do registru paměti.

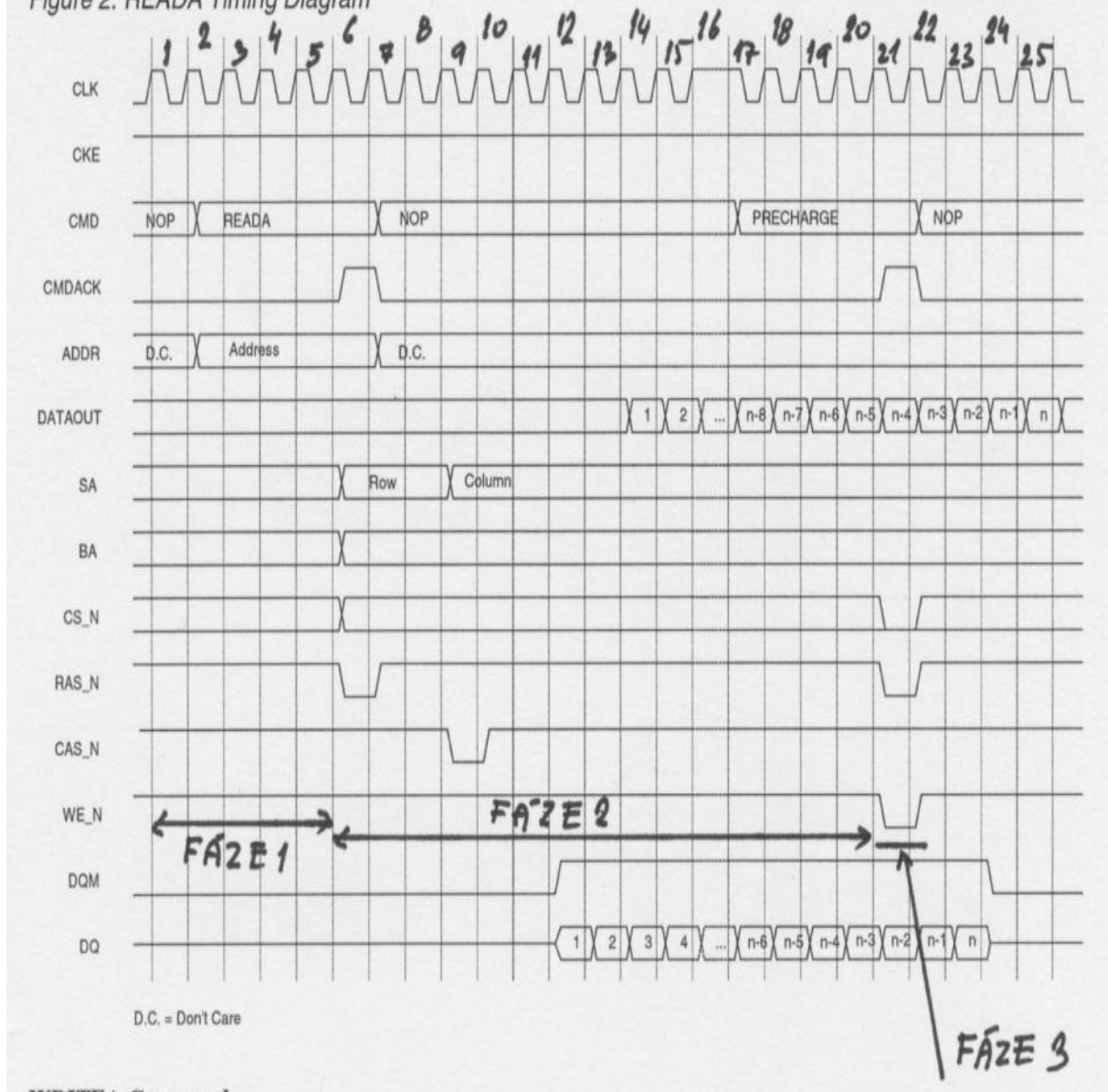
Při obou těchto činnostech má kombinace CAS, RAS, WE význam kódu příkazu, byť pomocí CAS a RAS se zároveň do SDRAM přenáší příslušná adresa.

Příkaz PCH (Precharge – uzavření banku): stav bitu SA[10] určuje, jestli má být příkaz PCH proveden na všech bankách nebo bank je určen bity BA[1:0].

Příkaz *LMR*: při provádění příkazu *Load mode register* je na vodičích SA operační kód tohoto příkazu: *Load register R1/Load register R2*.

- Jednotlivé banky SDRAM musí být před čtením či zápisem **otevřeny** – k tomu slouží příkaz *Activate (ACT)*. Součástí příkazu je adresace příslušného banku a řádku.
- Realizaci paměťové operace lze rozdělit na tři části:
  - **Fáze 1**: Komunikace mezi procesorem a řadičem paměti (končí generováním signálu *CMDACK* řadičem paměti směrem do nadřazené komponenty (procesoru).
  - **Fáze 2**: Komunikace mezi řadičem paměti a pamětí (začíná generováním příkazu řadičem paměti – signály *WE, CAS, RAS*).
  - **Fáze 3**: Ukončení komunikace – uzavření banků (příkaz *Precharge*).

Figure 2. READA Timing Diagram



## Fáze 1 (přenos kódu příkazu a adres do řadiče paměti):

1. Na vstup řadiče SDRAM je vložen kód příkazu (např. čtení) – signály CMD[2:0].

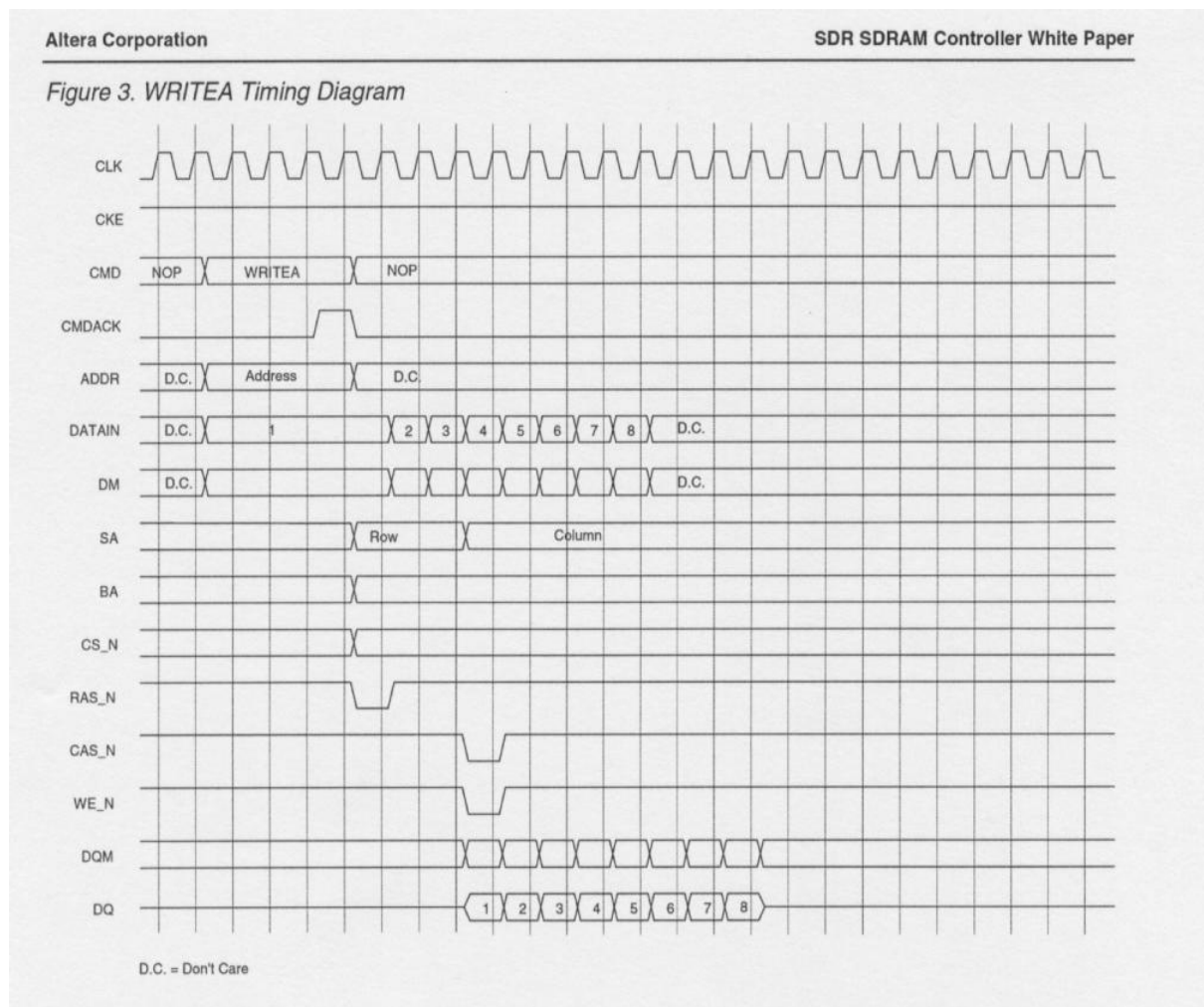
2. Řadič SDRAM příkaz přijme a potvrdí přijetí příkazu generováním signálu CMDACK (Command Acknowledge).
  3. Součástí této komunikace je předání adresy řádku (row) a sloupce (column) do řadiče paměti (vodiče ADDR).
  4. Na vstup řadiče SDRAM je vložen kód příkazu NOP – signály CMD[2:0] – s první hranou CLK po přijetí CMDACK.
  5. Platí, že pokud se od řadiče SDRAM a paměti nevyžaduje žádná činnost, pak je na CMD[2:0] příkaz NOP.
  6. Platí také, že každá změna v kódu příkazu je pamětí SDRAM rozpoznána a příkaz se provede.
- **Fáze 1 je realizována během prvních 5 synchronizačních pulsů. Na jejím konci je v řadiči uložena veškerá informace, kterou je potřeba pro provedení příkazu – kód příkazu a všechny potřebné adresy, je možné zahájit fázi 2.**
  - Fázi 1 považujeme za úvodní část komunikace, během níž se do paměti řadiče paměti posílá informace, která je důležitá pro realizaci vyžadované operace (**přípravná fáze**).

## Fáze 2 (přenos příkazu z řadiče SDRAM do SDRAM a jeho provedení):

- Provedení příkazu zadaného procesorem se rozdělí na několik kroků (příkazů).
- Jednotlivé banky SDRAM musí být před čtením či zápisem **otevřeny** – k tomu slouží příkaz *Activate* - *ACT*. Součástí příkazu je adresace příslušného banku a řádku.
- Příkaz *ACT* – kombinace RAS, CAS, WE podle Tab. 2, když je aktivní RAS, tak se zároveň přenáší adresa řádku (row) do SDRAM.
- Příkaz *ACT* je zahájen 6. synchronizačním pulsem, trvá do 8. synchronizačního pulsu.
- Příkaz *RD* – začíná 9. pulsem (Tab. 2), jeho součástí je přenos sloupce (column) do SDRAM a předání příkazu RD do SDRAM.
- Součástí tohoto příkazu je následně přenos dat z SDRAM do řadiče (signály DQ, maska DQM), následně pak přes DATAOUT z řadiče SDRAM do nadřazeného prvku (procesoru, disku).
- Fázi 2 můžeme označit za **prováděcí fázi** příkazu.

### **Fáze 3 (ukončení příkazu, uzavření banku):**

- Příkaz *Precharge PCH* – uzavření banku, ten je poslán do řadiče SDRAM, řadič SDRAM jej pošle do SDRAM až je jisté, že požadovaná data jsou získána (tzn. byl zahájen jejich přenos z SDRAM do řadiče SDRAM).
- Následuje příkaz **No operation** – Prázdňá operace.
- Fáze 3 – **ukončení příkazu.**





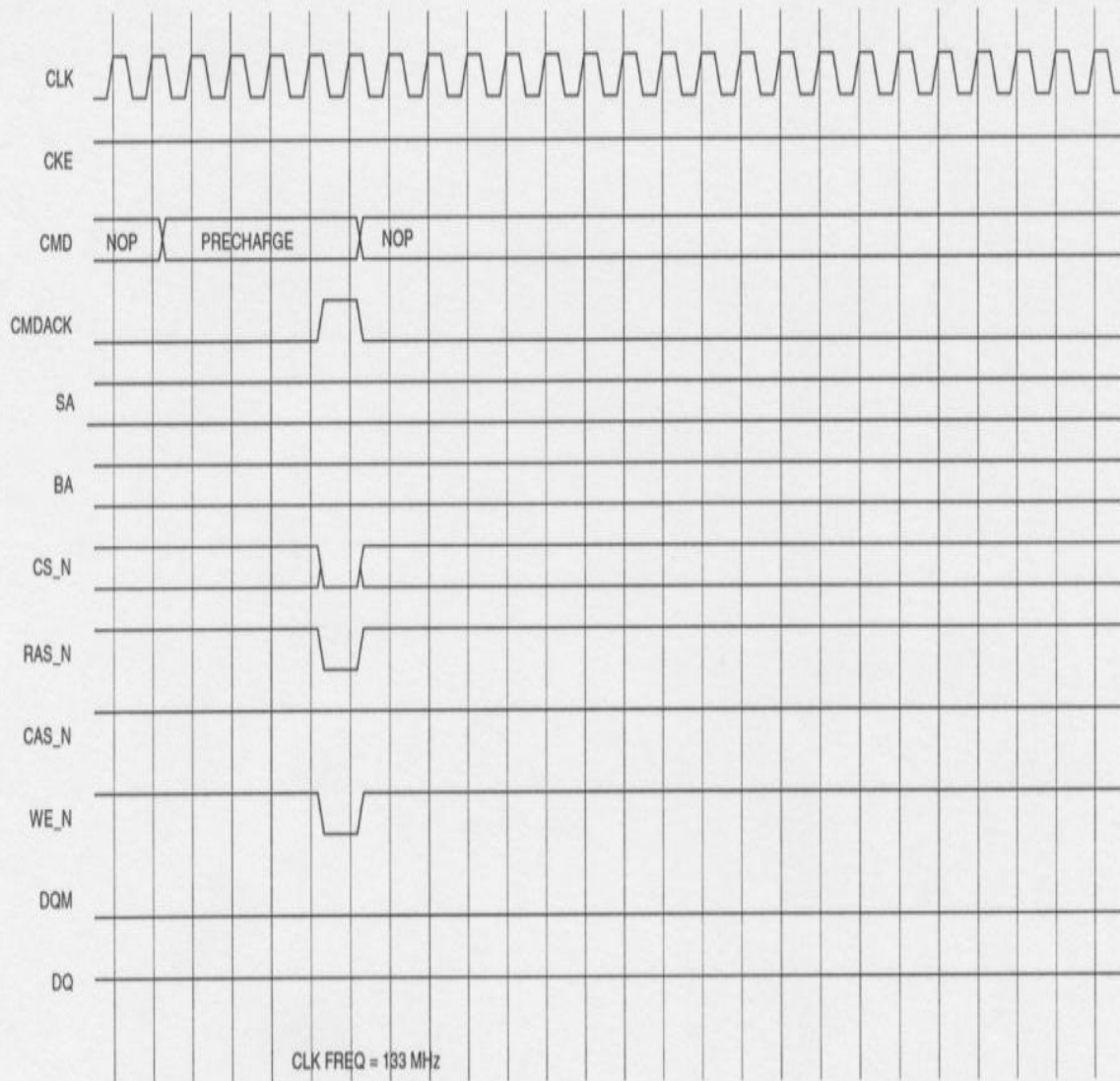
## **Zápis dat do paměti (příkaz WRITEA)**

- Stejně jako při čtení je možné i zde rozpoznat 3 fáze.
- Opačný směr přenosu dat: data je nutné napřed vložit do řadiče SDRAM, pak teprve do SDRAM.

## **Uzavření banku (příkaz PRECHARGE)**

- Posloupnost kroků:
  - Procesor vloží kód příkazu na signály CMD.
  - Řadič DMA generuje CMDACK (potvrzení převzetí signálu) a zahájí provádění příkazu vložení kódu příkazu na signály RAS, CAS, WE.
  - Procesor vloží na CMD příkaz NOP.

Figure 5. PRECHARGE Timing Diagram



## Příkaz Load mode register

- Důvody, proč musí být k dispozici *registr režimu* (mode register):

Existuje sestava počítače, do níž má být paměť zabudována.

V této sestavě existuje jistý synchronizační kmitočet, jímž má být komunikace mezi řadičem SDRAM a pamětí SDRAM řízen.

SDRAM má své dynamické parametry dané katalogovými listy (časové údaje, např. v  $\mu\text{s}$  – refresh), podle těchto parametrů musí být nastaven režim komunikace a jeho parametry.

- Informace, která se má zapisovat do registru režimu SDRAM, se z nadřazené komponenty vkládá do řadiče SDRAM z vodičů ADDR (vstup řadiče SDRAM). Hodnota na těchto signálech se přenáší na signály A0 - A11 (vstup SDRAM).
- Dva příkazy: Load\_REG1, Load\_REG2

- Význam bitů ADDR, příkaz Load\_REG1:

parametr	bity ADDR	Význam
CL	[1:0]	CAS latency
RCD	[3:2]	RAS to CAS delay in number of clocks
RRD	[7:4]	REFRESH command duration in clocks
PM	[8]	Režim řadiče SDRAM: 0 – normal, 1 – page mode
BL	[12:9]	Burst length, typické hodnoty 1, 2, 4, 8.

- RCD – zpoždění mezi RAS a CAS  

$$RCD = \text{INT}(t_{RCD}/\text{clock period})$$
 $t_{RCD}$  – hodnota z katalogu konkrétní SDRAM  
clock period – perioda synchronizačních pulsů SDRAM
- RRD – trvání příkazu REFRESH vyjádřené v počtu synchronizačních pulsů.  

$$RRD = \text{INT}(t_{RRD}/\text{clock\_period})$$
 $t_{RRD}$  je hodnota z katalogu SDRAM,  
clock\_period – perioda synchronizačních pulsů SDRAM

- Příkaz Load\_REG2:  
16 bitový registr, v němž je uložena hodnota, která reprezentuje periodu příkazů REFRESH, kterou bude dodržovat řadič SDRAM.

$INT(\text{refresh\_period}/\text{clock\_period})$

### **Příklad:**

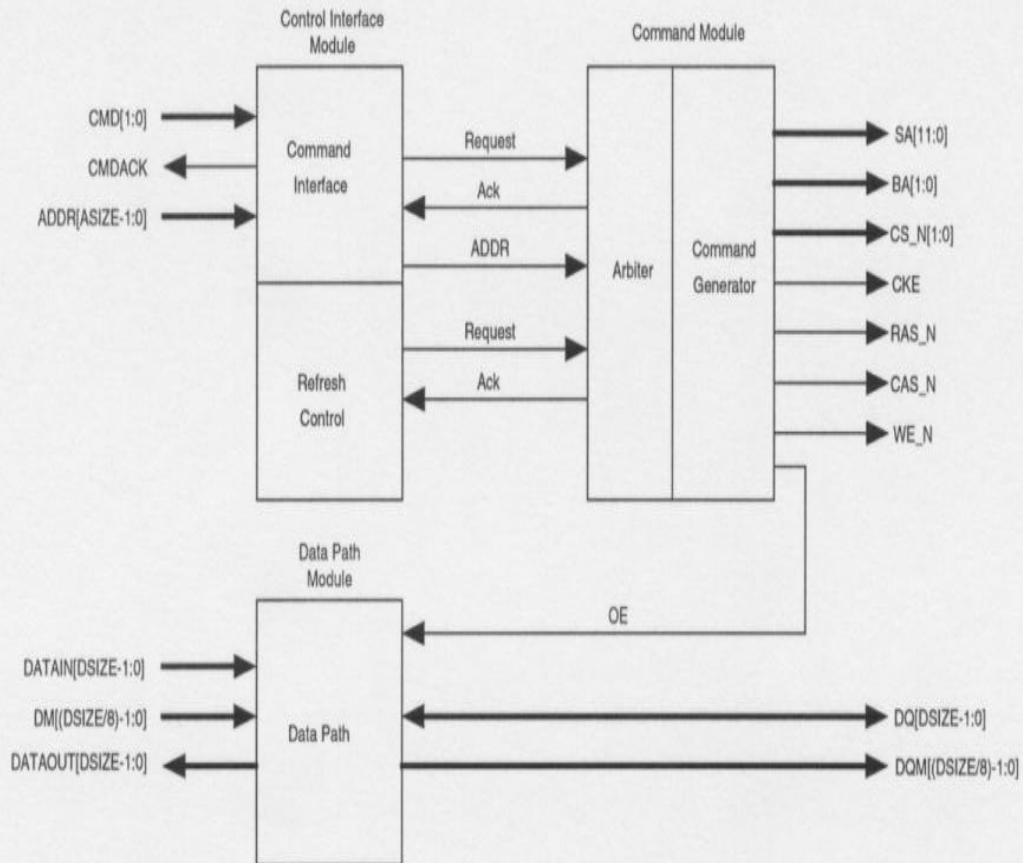
Refresh celé paměti trvá 64 ms, v rámci těchto 64 ms se musí provést 4096 cyklů (obojí údaje jsou katalogové), tzn. provedení příkazu Refresh musí být vyžadováno každých 15,62509  $\mu\text{s}$  ( $64 \text{ ms}/4096 = 15,62509 \mu\text{s}$ )

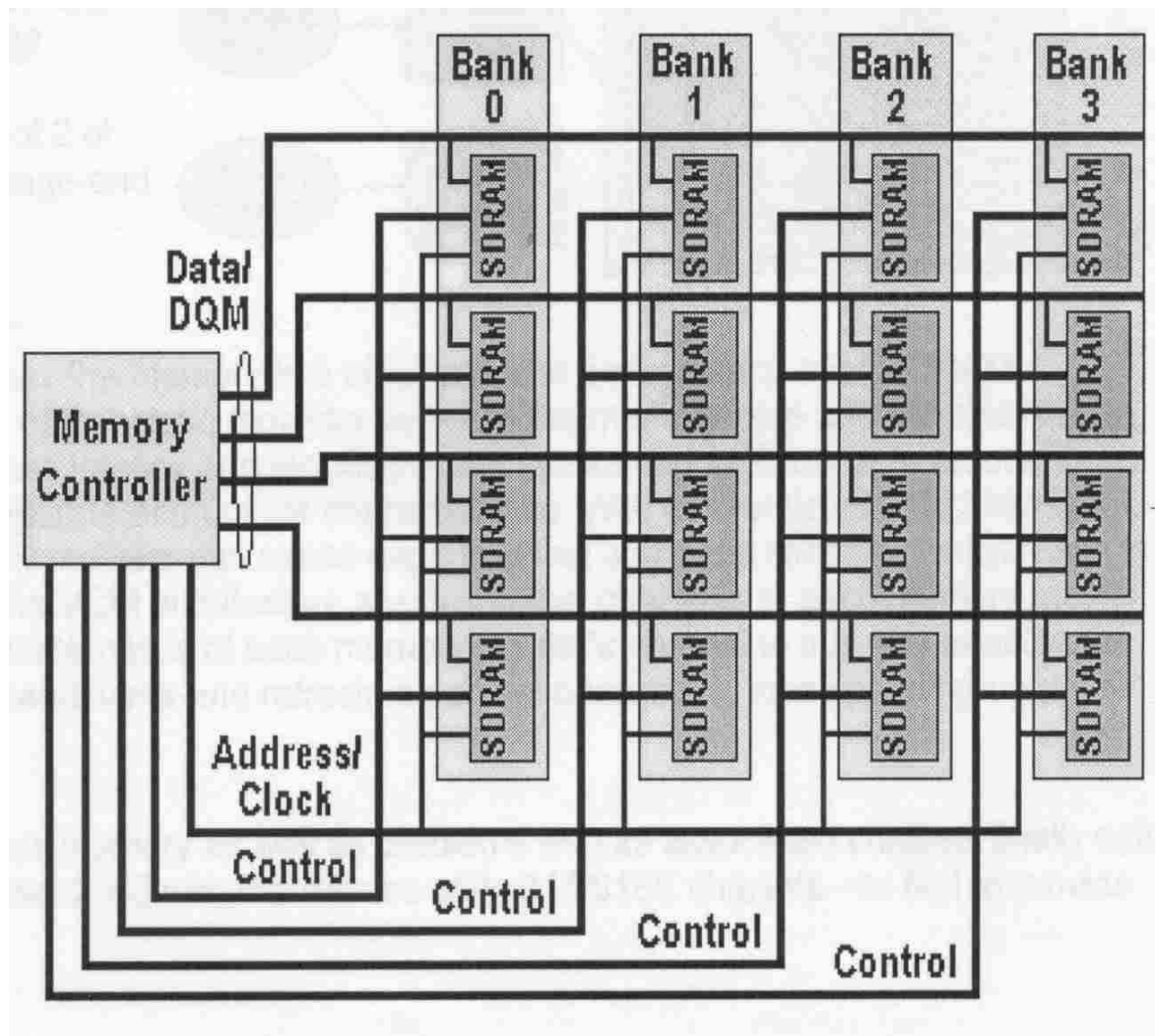
Pokud je řadič SDRAM a SDRAM synchronizován kmitočtem 100 MHz, pak počet synchronizačních pulsů, které musí být vygenerovány řadičem SDRAM mezi dvěma příkazy Refresh je 1562.

$$15,62509 \mu\text{s}/0.01 \mu\text{s} = 1562$$

- Hodnoty parametrů, které se zavádějí do registrů, jsou ve skutečnosti katalogové údaje (časové) převedené na počty synchronizačních signálů.

Figure 7. SDRAM Controller Block Diagram





## Single Data Rate (SDR DRAM)

- Přenosy se odehrávají pouze od jedné z hran synchronizačního signálu.

## Double Data Rate (DDR DRAM)

- Odlišnost od předcházejících typů: přenosové děje se odehrávají od nástupné i sestupné hrany synchronizačních pulsů - je tak ve skutečnosti

možné dvakrát zrychlit synchronizaci, aniž by se zvýšil kmitočet synchronizačních pulsů.

### **Další pojmy:**

- PC100 SDR SDRAM je navržena pro počítače se systémovou sběrnici s rychlostí 100 MHz (Pentium II, Pentium III, AMD K6-III, AMD Athlon, AMD Duron a systémy Power Mac G4).
- PC133 SDR SDRAM – pro počítače se systémovou sběrnici 133 MHz (Pentium III B, AMD Athlon a systémy Power Mac G4).
- PC1600 DDR SDRAM – pro počítače se systémovou sběrnici 100 MHz (200 MHz v DDR režimu). '1600' reprezentuje rychlost v b/s ( $200 \text{ MHz} \times 8 \text{ bitů} = 1,6 \text{ Gb/s}$ ). Později nahrazena plně kompatibilními prvky PC2100.
- Prvky PC2100 DDR SDRAM - pro počítače se systémovou sběrnici 133 MHz (266 MHz v režimu DDR). '2100' reprezentuje rychlost v b/s ( $266 \text{ MHz} \times 8 \text{ bitů} = 2,1 \text{ Gb/s}$ ) – počítače s procesory AMD Athlon, Pentium III, Pentium IV.
- Prvky PC2700 DDR SDRAM - pro počítače se systémovou sběrnici 166 MHz (333 MHz v režimu DDR). '2700' reprezentuje rychlost v b/s ( $333 \text{ MHz} \times 8 \text{ bitů} = 2,7 \text{ Gb/s}$ ).



- Prvky PC3200 DDR SDRAM - pro počítače se systémovou sběrnicí 200 MHz (400 MHz režimu DDR). '3200' reprezentuje rychlost v b/s ( $400 \text{ MHz} \times 8 \text{ bitů} = 3,2 \text{ Gb/s}$ ).